Exhibit H

Lu 7,437,583 Applied to Representative Panasonic and Toyota Accused Products

This claim chart compares independent claims 17 and 25 of U.S. Patent No. 7,437,583 ("the Lu '583 patent") to Texas Instruments' DRA750 system on a chip ("SoC").

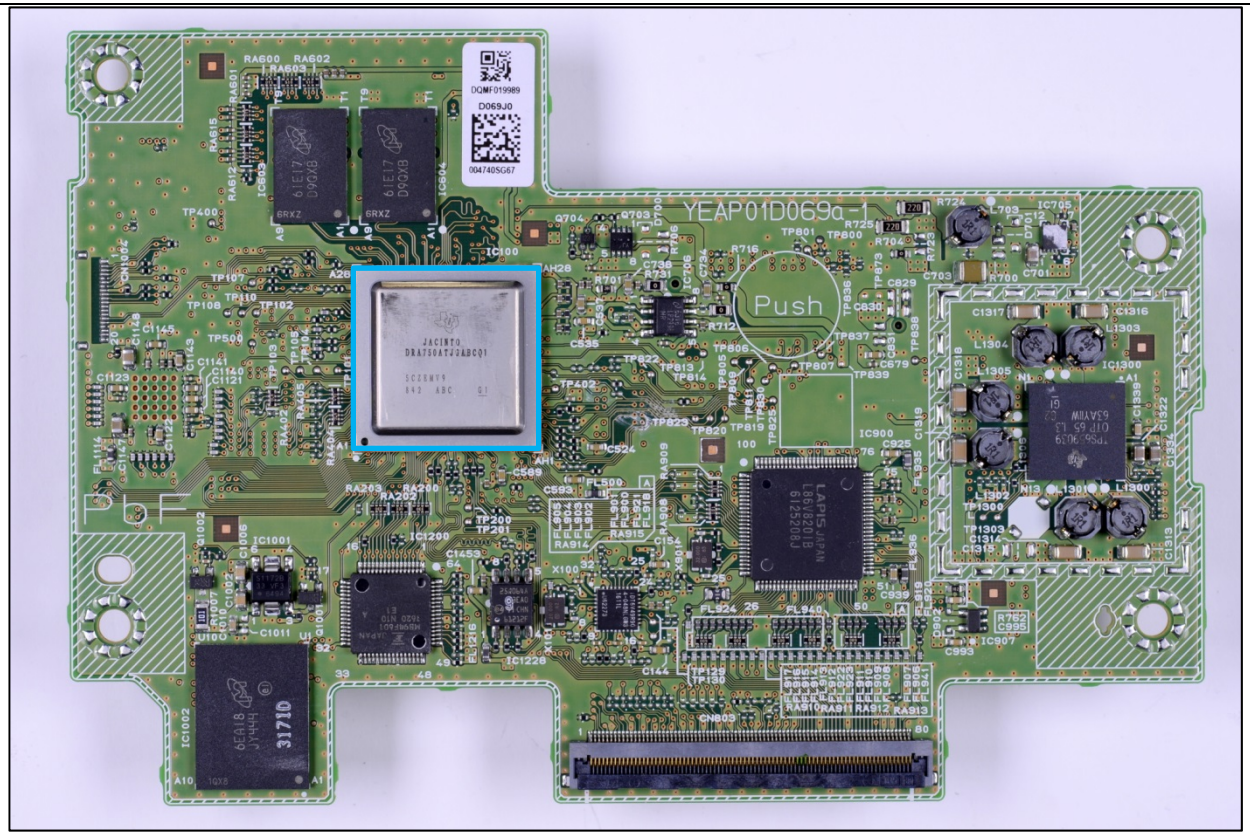
On information and belief, Texas Instruments' DRA750 SoC is representative of other Texas Instruments infotainment and high-end car information system SoCs having similar functionality ("Accused Texas Instrument Infotainment SoCs").

The DRA750 SoC is incorporated in downstream products, including without limitation, Panasonic head units, such as Ser. Nos. 130105, 104020, 104069, 500021, which are incorporated in Accused Toyota Navigation units, including Highlander Receiver (86804-0E280), Sienna Navigation Unit (86804-08040), Avalon Navigation Head Unit (86804-07120), and Prius III Navigation System Kit (86804-47330), respectively.

On information and belief the Accused Texas Instrument Infotainment SoCs, and head units and automobiles that incorporate the Accused Texas Instrument Infotainment SoCs infringe directly, indirectly, and or under the doctrine of equivalents, at least claims 17 and 25 of the Lu '583 patent.

Claims - U.S. Patent No. 7,437,583 (Lu)	**Application of Claim Language to Accused Products**
Claim 17	
A system for distributing clock signals within an electronic device, the system comprising:	To the extent that the preamble is deemed limiting, the TI DRA750 SoC and downstream products include a system for distributing clock signals within an electronic device. At least the Panasonic (AT1501) head unit, which is included in at least the Toyota Prius III Navigation System Kit (225202), includes a Texas Instruments DRA750 SoC (highlighted in blue).


Model No. 86804-47330
Ref.No. CQ-XS05E0BJ
Set No. AT1501
DC 12V
Negative Ground
Speaker 4 Ω
Panasonic Corporation
Made in Japan
Serial No. 500021
CLASS 1
LASER PRODUCT
MANUFACTURED
JULY 2016



The DRA750 includes a system for distributing clock signals within an electronic device.

3.1.1.1 Clock Management

The PRCM module manages the gating (that is, switching off) and enabling of the clocks to the device modules. The clocks are managed based on the requirement constraints of the associated modules. The following sections identify the module clock characteristics, management policy, clock domains, and clock domain management.

Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 409 (highlighted).

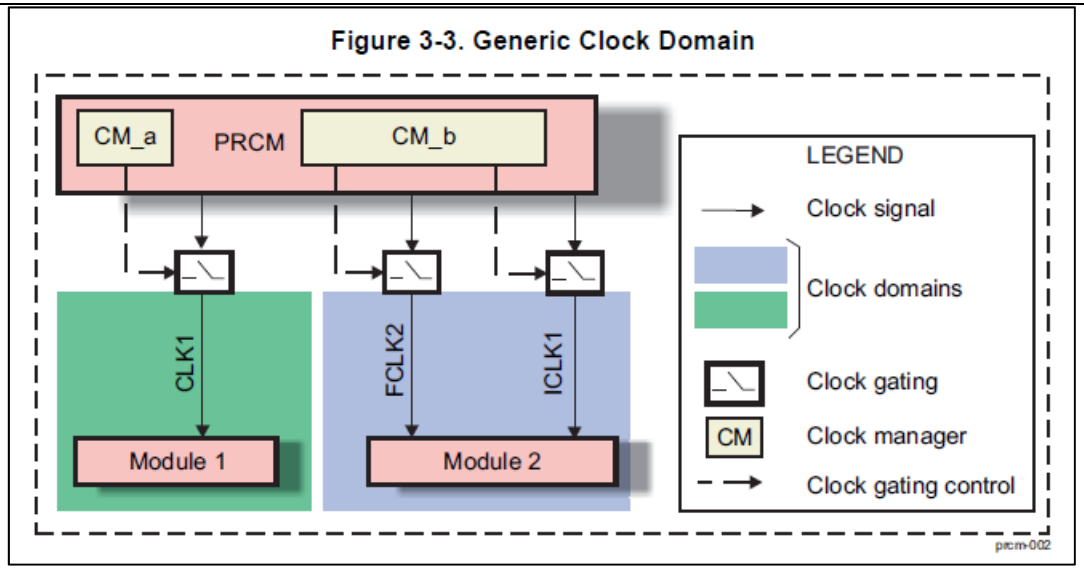
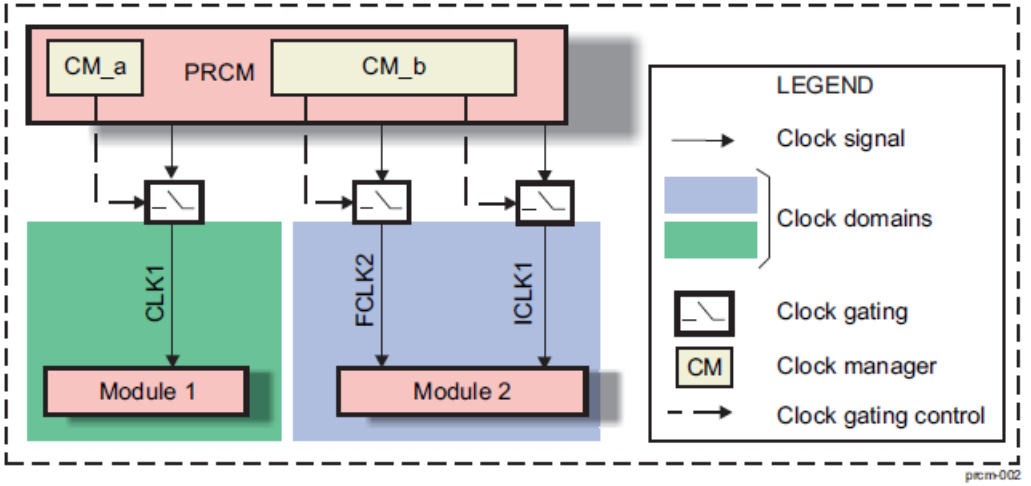
	 Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 415.
at least one processor that determines a status of at least one gate that controls flow of a clock signal to at least one device coupled to said at least one gate; and	The DRA750's system for distributing clock signals includes at least one processor that determines a status of at least one gate that controls flow of a clock signal to at least one device coupled to said at least one gate. The DRA750 includes at least one gate that controls flow of a clock signal to at least one device coupled to said at least one gate. For example, as shown below, the DRA750 includes "clock gating" gates that control the flow of the clock signal (solid line below) to at least one module/device coupled to the gate.

Figure 3-3. Generic Clock Domain



Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 415.

The DRA750 includes at least one processor that determines a status of at least one gate that controls flow of a clock signal to at least one device coupled to said at least one gate. For example, the DRA750's power, reset, and clock management ("PRCM") module is a processor that determines a status of at least one gate using, for example, the "CLOCKACTIVITY" bit.

3.1.1.1.2 Module-Level Clock Management

Each module in the device may also have specific clock requirements. Certain module clocks must be active when operating in specific modes, or they may be gated. Globally, the activation and gating of the module clocks are managed by the PRCM module. Hence, the PRCM module must be aware of when to activate and when to gate the module clocks.

v

Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 410.

Table 3-6. Slave Module Clock Activity Settings

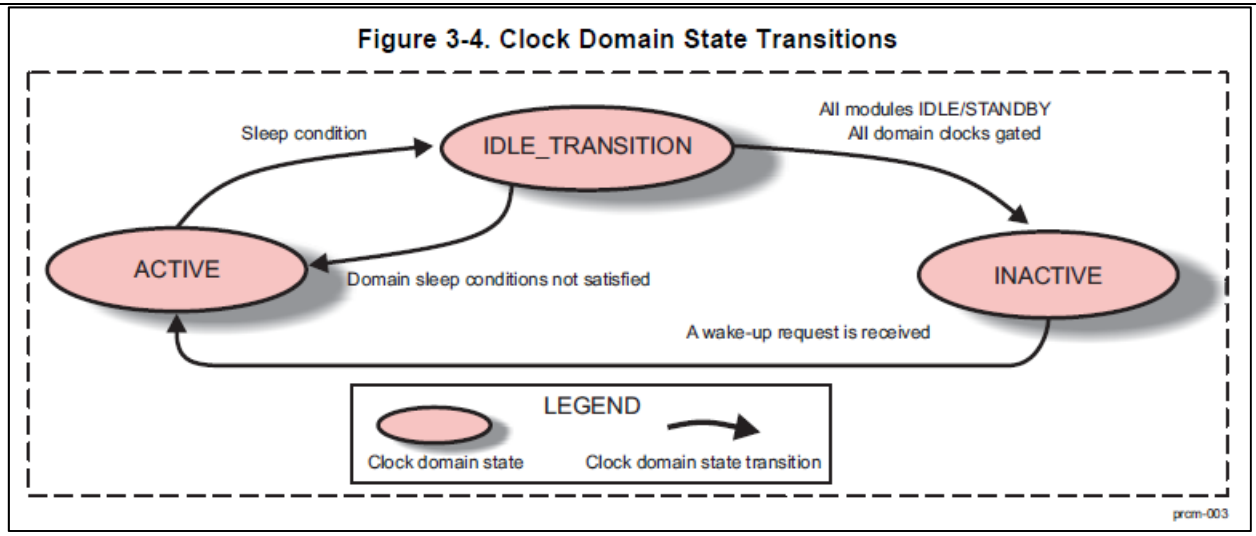
CLOCKACTIVITY Bit Value	Module Interface Clock	Module Functional Clock	Description
0x0	Gated	Gated	The interface and functional clocks are considered when generating the acknowledgment. This setting also means both clocks may be gated upon a PRCM module IDLE request.
0x1	Active	Gated	The interface clock is not shut down and is not considered when generating the acknowledgment to the PRCM module IDLE request. Only the functional clock is considered.
0x2	Gated	Active	The functional clock is not shut down and is not considered when generating the acknowledgment to the PRCM module IDLE request. Only the interface clock is considered.
0x3	Active	Active	The interface and functional clocks are not shut down. The module can acknowledge the IDLE request without checking the internal functions linked to its clocks.

Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 413.

> The PRCM module does not have any hardware means to read the CLOCKACTIVITY settings of the module. Software must ensure consistent programming between the CLOCKACTIVITY settings of the module and the clock-gating control bits in the PRCM module. The PRCM module must be configured (where software control is available) not to gate the module clock, which should remain active according to the CLOCKACTIVITY settings of the module.

Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 413.

By way of another example, the PRCM module determines a status of at least one gate that controls flow of a clock signal to at least one device coupled to said at least one gate by determining that all domain gates are gated when transitioning to an inactive state.

	 Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 416.
said at least one processor controls said at least one gate based on said determined status.	The DRA750's at least one processor controls said at least one gate based on said determined status. For example, the DRA750's PRCM module controls the gate based on the status of the "CLOCKACTIVITY" bit. The PRCM module does not have any hardware means to read the CLOCKACTIVITY settings of the module. Software must ensure consistent programming between the CLOCKACTIVITY settings of the module and the clock-gating control bits in the PRCM module. The PRCM module must be configured (where software control is available) not to gate the module clock, which should remain active according to the CLOCKACTIVITY settings of the module. Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 413 (highlighted).

The PRCM module lets software check the status of the clock domain functional clocks. The CM_<Clock domain>_CLKSTCTRL[x] CLKACTIVITY_FCLK/<Clock name>_FCLK bit in the PRCM module identifies the state of the functional clock(s) within the clock domain. Table 3-10 lists the two possible states of the functional clock.

Table 3-10. Clock Domain Functional Clock States

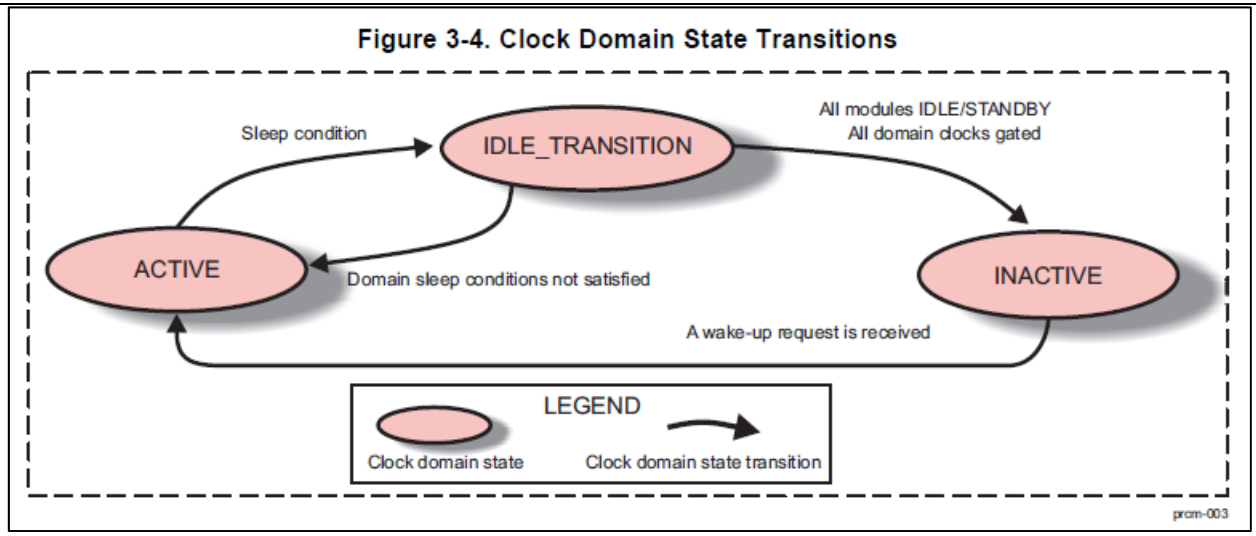
CLKACTIVITY Bit Value	Status	Description
0x0	Gated	The functional clock of the clock domain is inactive.
0x1	Active	The functional clock of the clock domain is running.

Table 3-11. Clock Domain Interface Clock States

CLKACTIVITY Bit Value	Status	Description
0x0	Gated	The interface clock of the clock domain is inactive.
0x1	Active	The interface clock of the clock domain is running.

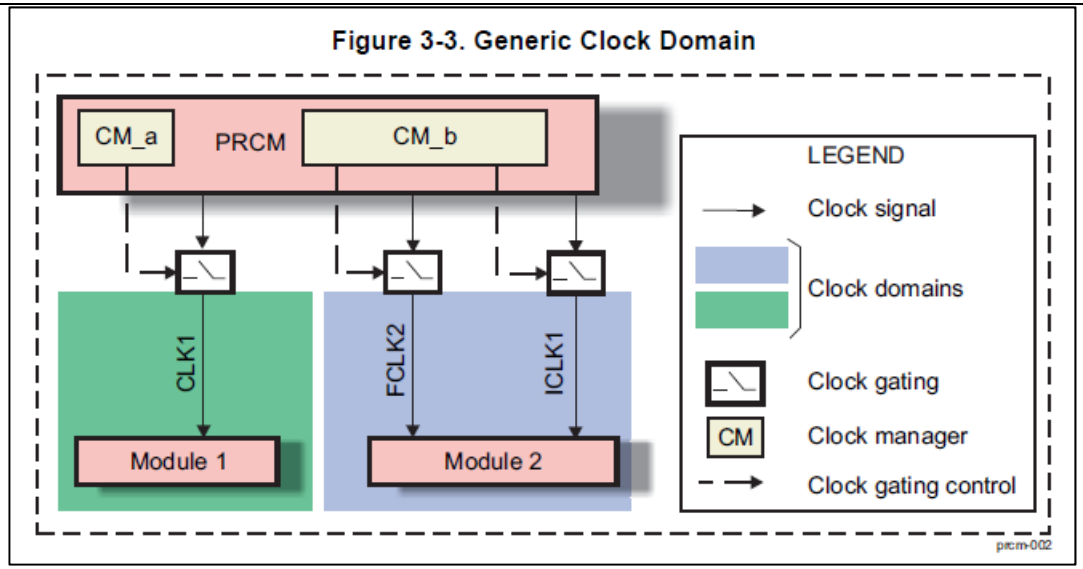
Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 416.

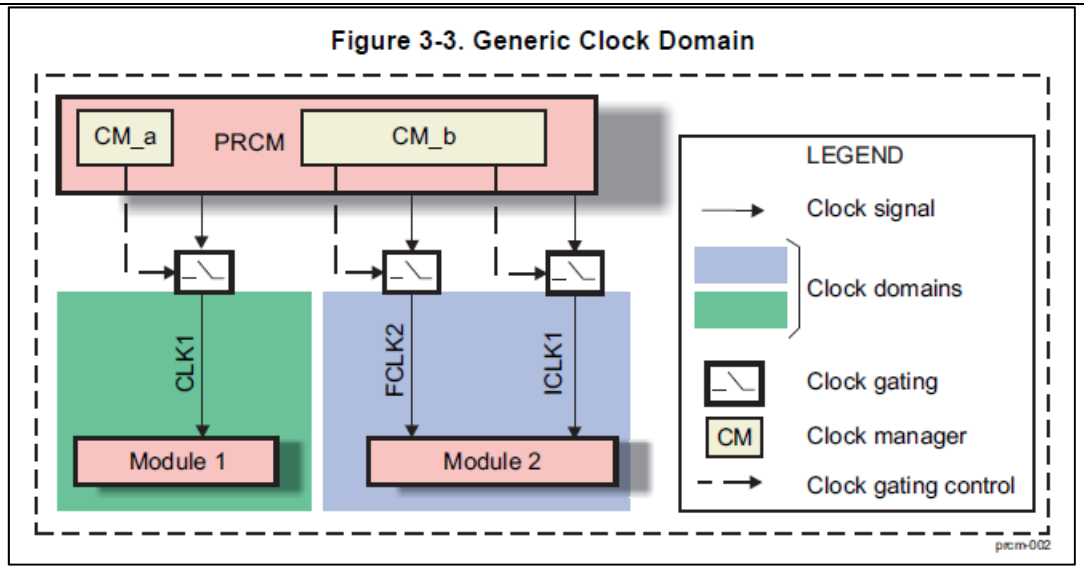
By way of another example, the DRA750's PRCM module controls the clock domain gates based on the status of the gates (e.g., "All domain clocks gated"). As the figure below shows, a transition from "INACTIVE," where the clock gates are gated, to "ACTIVE," where the clock gates are active, requires: (1) "A wake-up request is received" and (2) a determination that "All modules IDLE/STANDBY" and "All domain clocks gated."

<table>
<tr><td></td><td> 

Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 416.</td></tr>
<tr><td>Claim 25</td><td></td></tr>
<tr><td>A system for distributing clock signals within an electronic device, the system comprising:</td><td>The DRA750 includes a system for distributing clock signals within an electronic device.

3.1.1.1 Clock Management
The PRCM module manages the gating (that is, switching off) and enabling of the clocks to the device modules. The clocks are managed based on the requirement constraints of the associated modules. The following sections identify the module clock characteristics, management policy, clock domains, and clock domain management.
Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 409 (highlighted).</td></tr>
</table>

	 Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 415.
a clock tree having a plurality of gates;	The DRA750 includes a clock tree having a plurality of clock gates. For example, as shown below, each clock domain includes at least one gate. The DRA750 includes 30 clock domains. Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 544-678.

	 Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 415.
a hardware control logic block coupled to said clock tree that controls at least a portion of said plurality of gates;	The DRA750 includes a hardware control logic block coupled to said clock tree that controls at least a portion of said plurality of gates. For example, the PRCM system includes at least one hardware control logic block coupled to said clock tree that controls at least a portion of said plurality of gates. ***3.1.1.1.2 Module-Level Clock Management*** Each module in the device may also have specific clock requirements. Certain module clocks must be active when operating in specific modes, or they may be gated. Globally, the activation and gating of the module clocks are managed by the PRCM module. Hence, the PRCM module must be aware of when to activate and when to gate the module clocks. The PRCM module differentiates the clock-management behavior for device modules based on whether the module can initiate transactions on the device interconnect (called master module) or it cannot initiate transactions and only responds to the transactions initiated by the master (called slave module). Thus, two hardware-based power-management protocols are used: • Master standby protocol: Clock-management protocol between the PRCM and master modules • Slave idle protocol: Clock-management protocol between the PRCM and slave modules Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 410 (highlighted).

	Master standby protocol This protocol is used to indicate that a master module must initiate a transaction on the device interconnect and requests specific (functional and interface) clocks for that purpose. The PRCM module ensures that the required clocks are active when the master module requests the PRCM module to enable them. This is called a module wake-up transition and the module is said to be functional after this transition completes. Similarly, when the master module no longer requires the clocks, it informs the PRCM module, which can then gate the clocks to the module. The master module is then said to be in standby mode. Although the protocol is completely hardware-controlled, software must configure the clock-management behavior for the module. This is done by setting the <*Module*>_SYSCONFIG. MIDLEMODE or <*Module*>_SYSCONFIG. STANDBYMODE bit fields, as described in Table 3-1. The behavior, identified in the STANDBYMODE Bit Value column, must be configured. Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 410 (highlighted). **Slave idle protocol** This hardware protocol allows the PRCM module to control the state of a slave module. The PRCM module informs the slave module, through assertion of an IDLE request, when its clocks (interface and functional) can be gated. The slave can then acknowledge the request from the PRCM module, and the PRCM module is then allowed to gate the clocks to the module. A slave module is said to be in IDLE state when its clocks are gated by the PRCM module. Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 411 (highlighted).
at least one register that is controlled by a clock tree driver; and	The DRA750 includes at least one register that is controlled by a clock tree driver. For example, clock tree driver software running on a processor sets the “CLKTRCTRL” bit register to perform a software-forced wakeup of the clock domain clocks.

<table>
<tr><td></td><td>

Table 3-13. Clock Domain Clock Transition Mode Settings

CLKTRCTRL Bit Value	Selected Mode	Description
0x0	NO_SLEEP	A clock domain sleep transition is never initiated, regardless of the hardware conditions.
0x1	SW_SLEEP	A software-forced sleep transition. The transition is initiated when the associated hardware conditions are satisfied (see Table 3-15).
0x2	SW_WKUP	A software-forced clock domain wake-up transition is initiated, regardless of the hardware conditions identified in Table 3-14.
0x3	HW_AUTO	Hardware-controlled automatic sleep and wake-up transition is initiated by the PRCM module when the associated hardware conditions are satisfied (see Table 3-14 and Table 3-15).

Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 417.
</td></tr>
<tr><td>at least one processor that overwrites a status of at least a portion of said plurality of gates which is controlled by said hardware control logic block.</td><td>

The DRA750 includes at least one processor that overwrites a status of at least a portion of said plurality of gates which is controlled by said hardware control logic block.

For example, at least one processor sets the “CLKTRCTRL” bit register to perform a software-forced wakeup of the clock domain clocks, which can overwrite the gated status of the clock domain’s clocks. Upon information and belief, the one processor is located within the PRCM or one of the Cortex-5 processors in the DRA750.

Table 3-13. Clock Domain Clock Transition Mode Settings

CLKTRCTRL Bit Value	Selected Mode	Description
0x0	NO_SLEEP	A clock domain sleep transition is never initiated, regardless of the hardware conditions.
0x1	SW_SLEEP	A software-forced sleep transition. The transition is initiated when the associated hardware conditions are satisfied (see Table 3-15).
0x2	SW_WKUP	A software-forced clock domain wake-up transition is initiated, regardless of the hardware conditions identified in Table 3-14.
0x3	HW_AUTO	Hardware-controlled automatic sleep and wake-up transition is initiated by the PRCM module when the associated hardware conditions are satisfied (see Table 3-14 and Table 3-15).

Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 417.
</td></tr>
</table>

Table 3-14. Clock Domain Wake-Up Conditions

Relation	Condition
OR	The SW_WKUP clock transition mode for the clock domain is set (CLKTRCTRL = 0x2).
	At least one wake-up request is asserted by one of the modules of the clock domain.
	At least one dynamic dependency(1) from another clock domain is active.
	At least one static dependency(1) from another clock domain is active.

Ex. 65 - DRA75x, DRA74x Technical Reference Manual at 420.